Physically Based Rendering

Usage of Physics in Rendering

Introduction

- Software Engineer @ Solid Angle
  - Working on the Maya pipeline
  - Tools, shaders for the users
  - Generally trying to improve Rendering and Lighting
  - GPU (OpenGL so far ☹)
- Previously @ Digic Pictures
  - Shaders, Rendering, Lighting and GPU ray tracing
  - Replacing the Studio’s old shaders with PBR shaders

Introduction

- What are we doing at Solid Angle?
  - Providing the industry with a fast, robust un-biased ray tracer that can handle huge amount of geometric complexity
  - Ten thousands of lights
  - Trillions polygons
  - Terrabytes of textures
- Like …

WHISKYTREE

Image courtesy of Digic Pictures. © 2013 Ubisoft Entertainment. All Rights Reserved. Watch Dogs, Ubisoft and the Ubisoft logo are trademarks of Ubisoft Entertainment in the US and/or other countries.

Introduction

- Why is it interesting
  - Rendering is cool ☺
  - Movies and Games
  - Explosion!

Introduction

- But seriously…
  - Shows how simplifying the inputs and controls can lead to a more controllable and realistic “simulation”
  - Materials, objects are behaving the same way under many different lighting conditions
  - Taking away control from users is always good
  - You can get great results even when following simple rules
    - What we can make even more complex in the future involving more and more physics

The Dark Ages

- The past
  - Not enough computing power led to simple lighting models
  - Lots of tweaking to materials
  - Nothing is based on physical properties (IOR for example)
  - Materials could emit more light than they actually received
  - Very hard to make it look realistic

The Dark Ages

- VFX
  - Since VFX studios got more computing power quicker than games
    - Age of highly complex, uber shaders that do everything
    - Too many controls and output data
  - Fix it in post!
    - Trying to fix 3D renders in 2D space
    - Confusion and every shot requires a huge amount of tweaking
    - Manually painting over the images
    - Correctness and realism depends on the Compositor

The Dark Ages

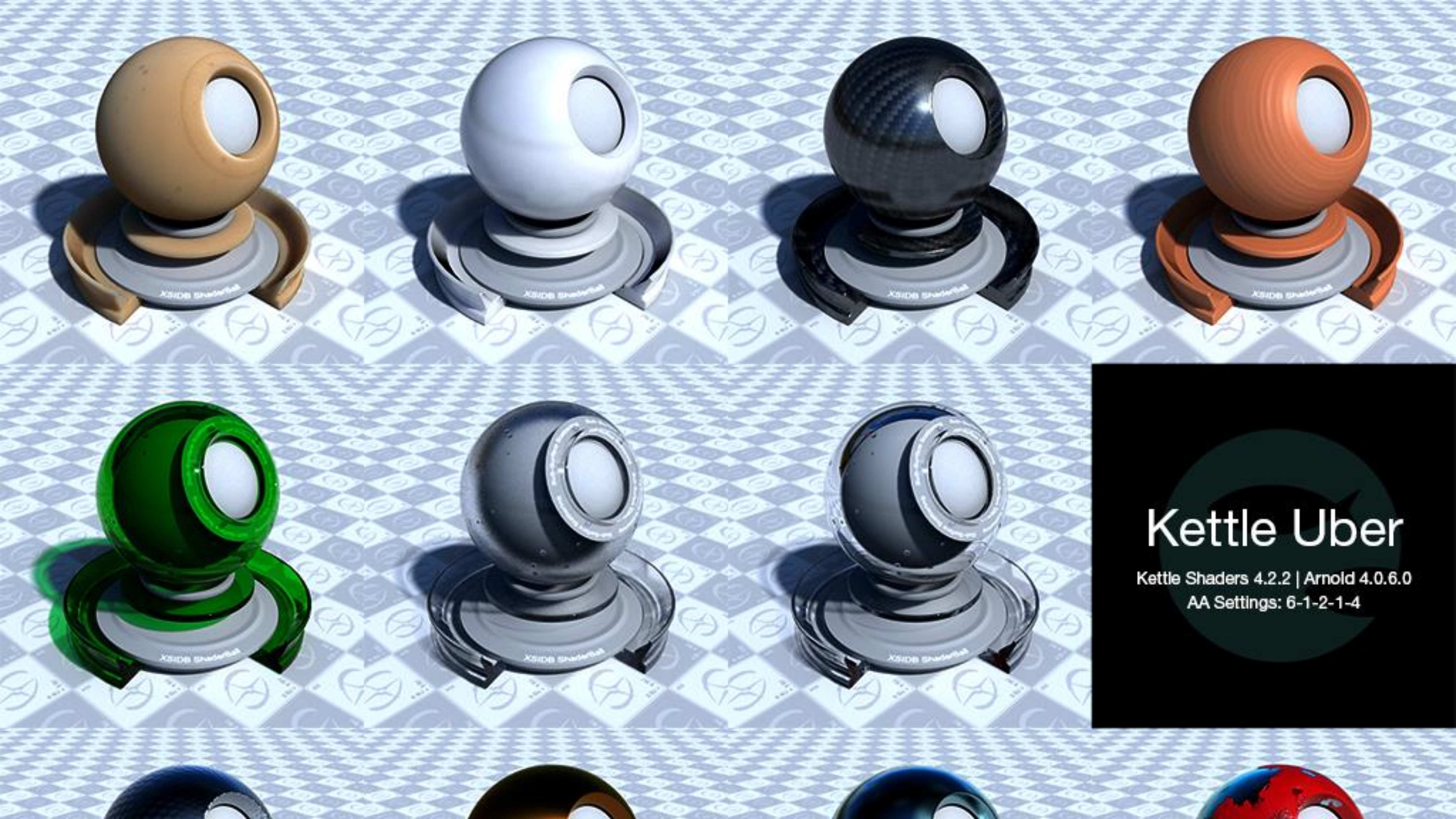
- Why is it a bad approach
  - Takes too much resources to fix everything
  - Hard to replicate a real world object
  - Complex shaders mean slow renders

Holy grail

- When did it start?
  - The first / second Iron Man @ ILM
    - They saw the need for a change
    - Huge amount of metallic surfaces that are hard to do otherwise
  - Physically Based Rendering Theory (Matt Pharr and Greg Humphrey - 2004)

Holy grail

- Arnold
  - Simple controls
  - Lots of user made PBR shaders (Kettle, alShaders, Gecko)
  - We are cheating though, our own “standard” shader is awful, easy to break it
- Unreal Engine 4
  - And basically every other modern game engine these days
- And the results …



Kettle Uber

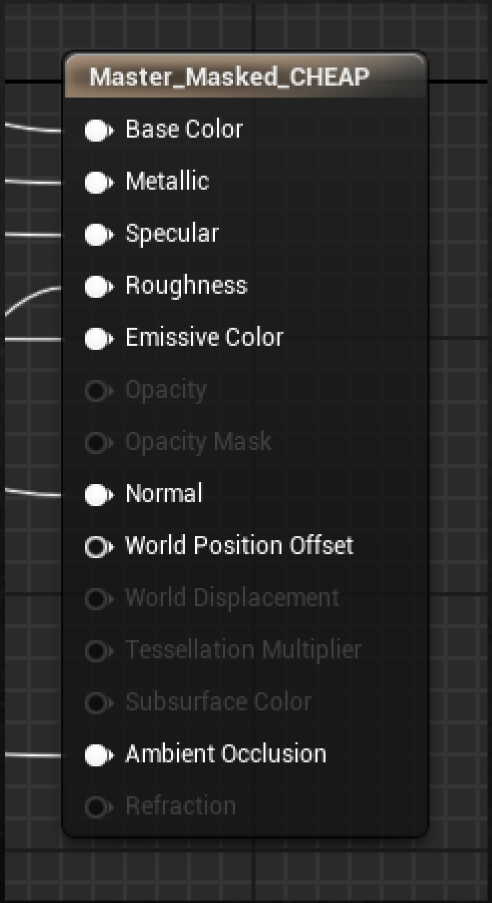
Kettle Shaders 4.2.2 | Arnold 4.0.6.0
AA Settings: 6-1-2-1-4

SERVICE HATCH
SERVICE HATCH
TEST SINGLE MOTOR
STEP
DISPLAY
ICE COLD
RELAY TEST
STEP
DISPLAY
ICE COLD
COIN TUBE FILL
STEP
DISPLAY
ICE COLD
IN CASE OF FIRE
BREAK GLASS

ALL YOU NEED IS RICK IS ALL YOU NEED
EPIC GAMES
23
MONEY BAG

Results

- Look way better (more realistic) than the older techs
- Simpler base shaders

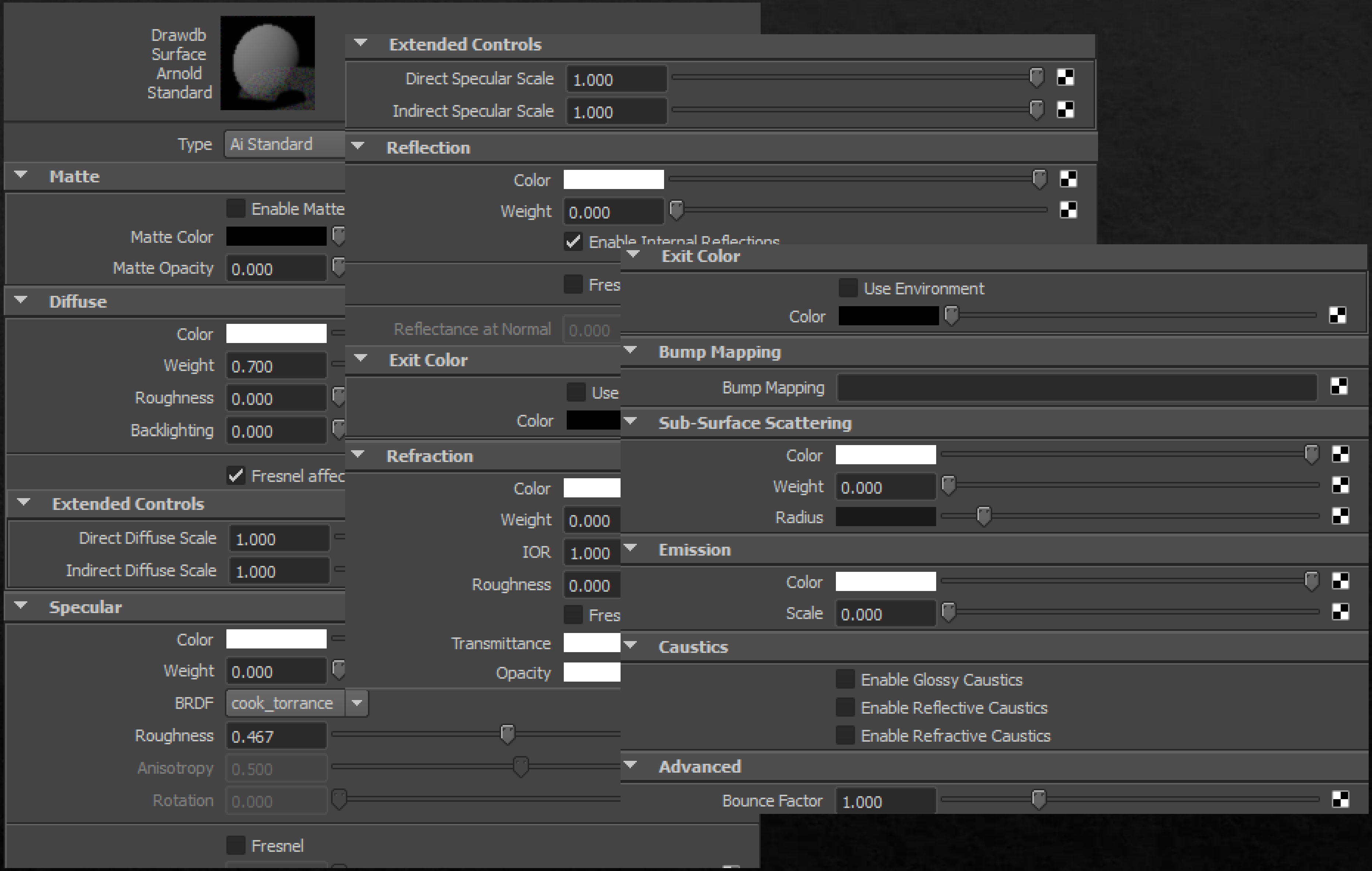


Master_Masked_CHEAP

- ● Base Color
- ● Metallic
- ● Specular
- ● Roughness
- ● Emissive Color
- ○ Opacity
- ○ Opacity Mask
- ● Normal
- ○ World Position Offset
- ○ World Displacement
- ○ Tessellation Multiplier
- ○ Subsurface Color
- ● Ambient Occlusion
- ○ Refraction

11

VS



Drawdb Surface Arnold Standard

Type: Ai Standard

Matte

□ Enable Matte

Matte Color

Matte Opacity 0.000

Diffuse

Color

Weight 0.700

Roughness 0.000

Backlighting 0.000

☑ Fresnel affec

Extended Controls

Direct Diffuse Scale 1.000

Indirect Diffuse Scale 1.000

Specular

Color

Weight 0.000

BRDF cook_torrance

Roughness 0.467

Anisotropy 0.500

Rotation 0.000

□ Fresnel

Extended Controls

Direct Specular Scale 1.000

Indirect Specular Scale 1.000

Reflection

Color

Weight 0.000

☑ Enable Internal Reflections

□ Fres

Reflectance at Normal 0.000

Exit Color

□ Use

Color

Refraction

Color

Weight 0.000

IOR 1.000

Roughness 0.000

□ Fres

Transmittance

Opacity

Exit Color

□ Use Environment

Color

Bump Mapping

Bump Mapping

Sub-Surface Scattering

Color

Weight 0.000

Radius

Emission

Color

Scale 0.000

Caustics

□ Enable Glossy Caustics

□ Enable Reflective Caustics

□ Enable Refractive Caustics

Advanced

Bounce Factor 1.000

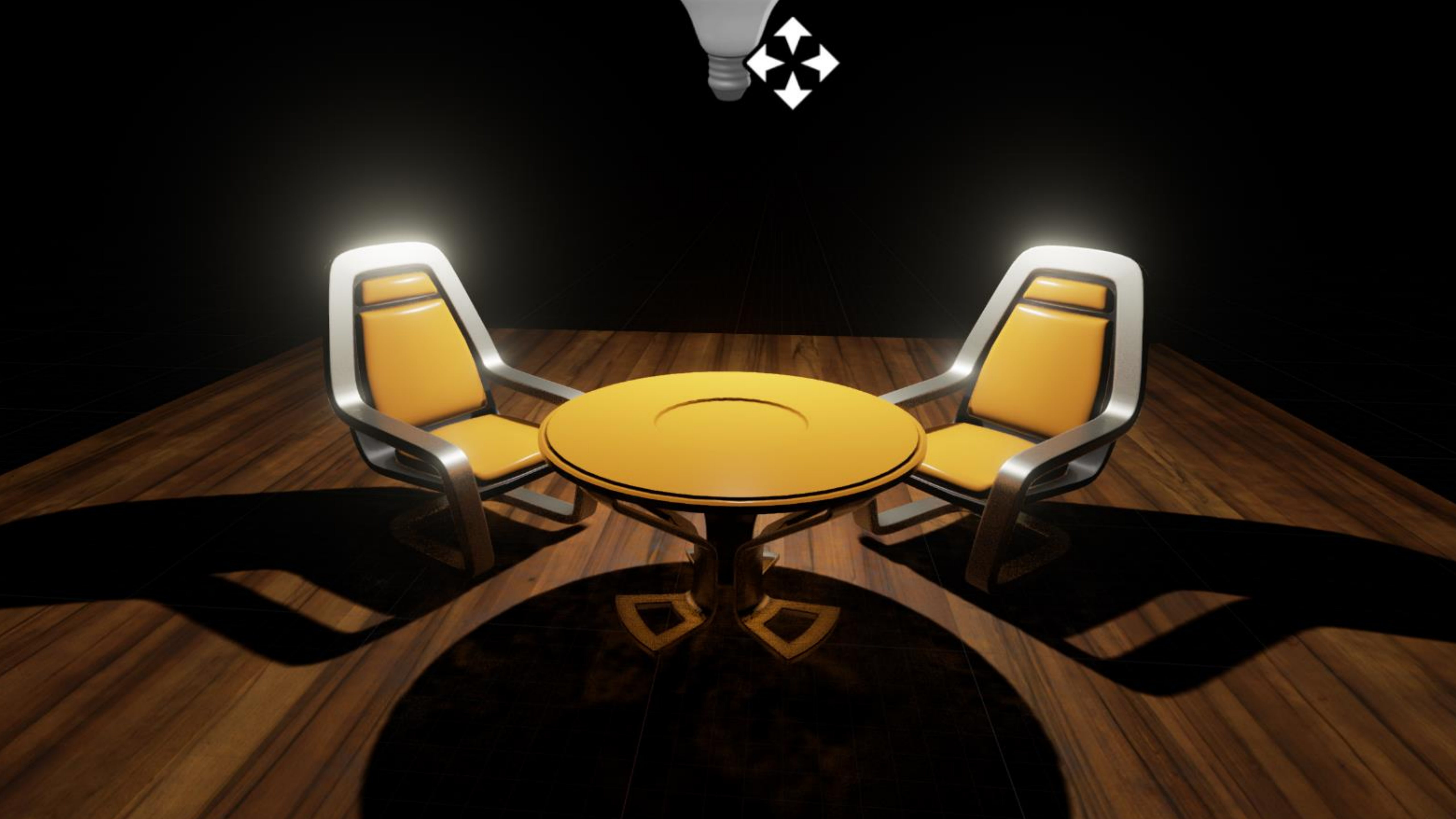
17 33 45

Results

- Look way better than the older techs
- Simpler base shaders
- Renders pretty fast (60FPS @ 4K on a Modern GPU)

In a nutshell

- Lights
  - Inverse Square Falloff



In a nutshell

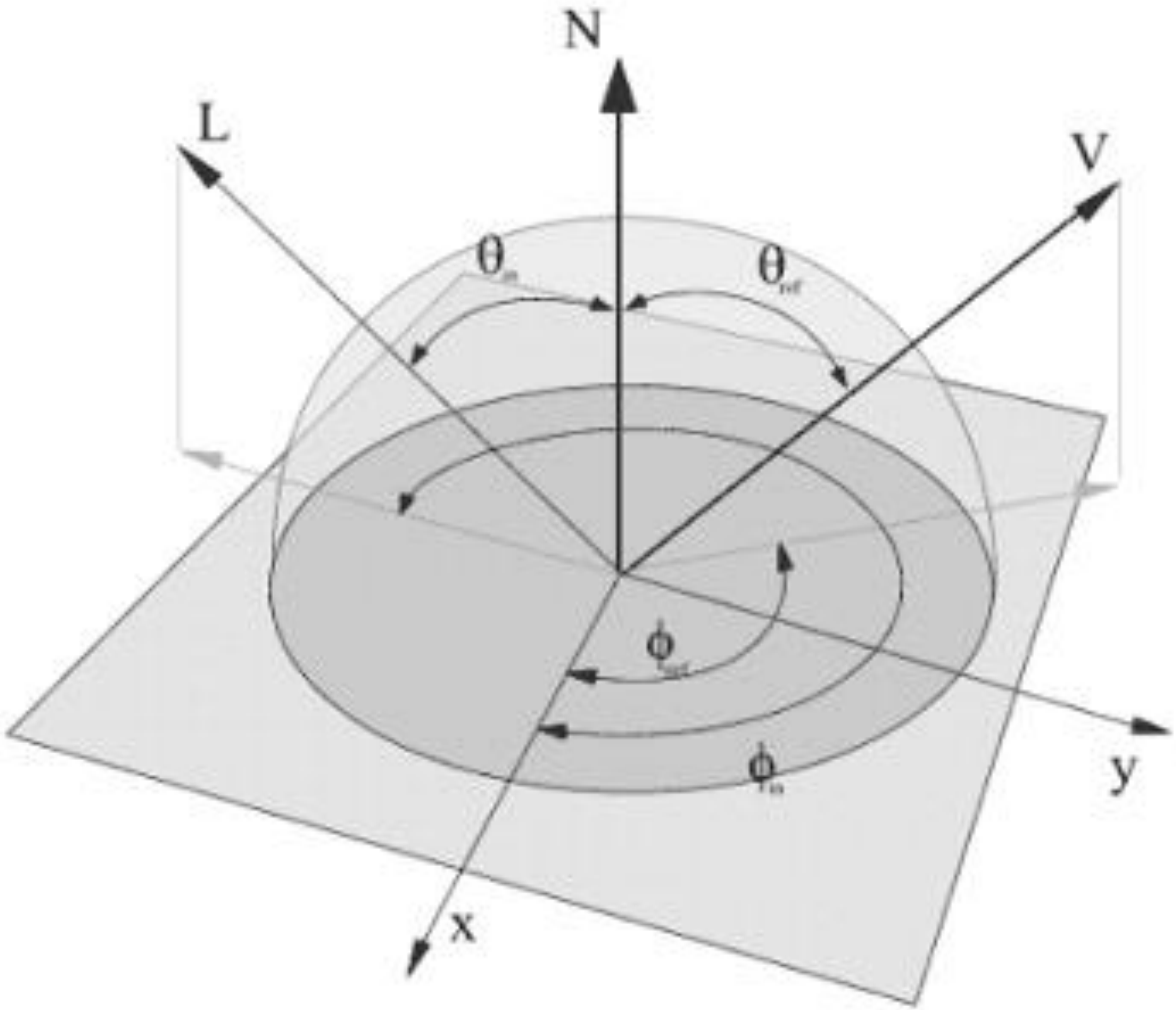
- Lights
  - Inverse Square Falloff
  - Every light is an area light

arnold
arnold
a

arnold
arnold

In a nutshell

- Materials
  - No material should reflect more light than received
    - Use some really nice, energy conserving BRDFs



N
L
V
θ_{in}
θ_{ref}
ϕ_{ref}
ϕ_{in}
y
x

In a nutshell

- Materials
  - No material should reflect more light than received
    - Use some really nice, energy conserving BRDFs
  - IOR and k based controls for Fresnel
    - Sometimes simpler, non physical controls are used, that can be mapped between 0 and 1. Textures!
  - No separate reflection and specular
    - Just say no!
  - Everything is glossy! (they are just sharp sometimes)

In a nutshell

- Cameras
  - Really important, often overlooked
  - Real world controls
    - Film Gate
    - Focal length
    - F-number
    - Shutter speed
    - Latency
    - Film Speed (ISO)
    - While Balance
    - Etc..
  - Not part of the big game engines by default (neither in Arnold)

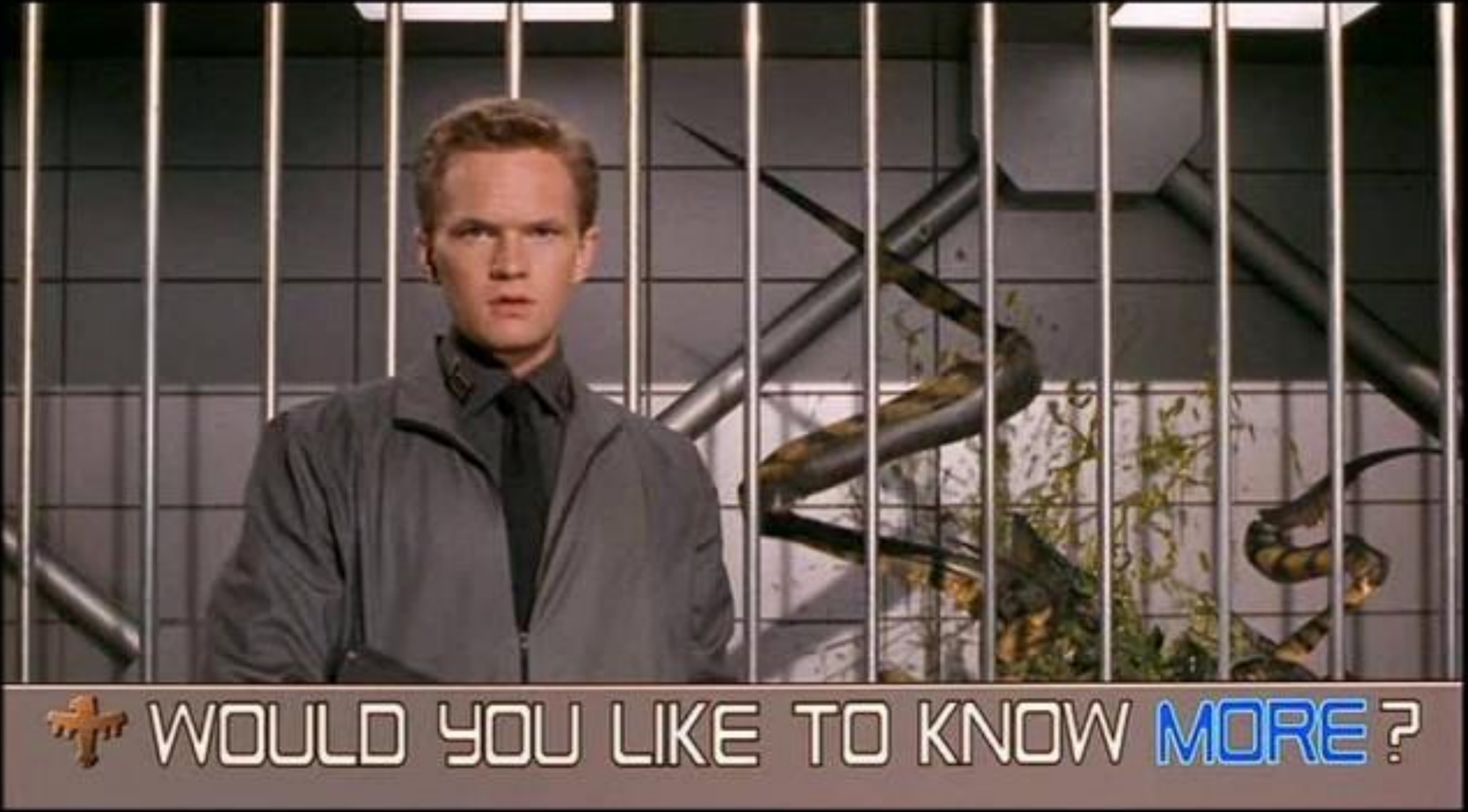
No 932188.
10
5
2

Future Work

- Global Illumination for everyone!
  - Common in VFX, animation, Games are still jealous

Future Work

- Global Illumination for everyone!
  - Common in VFX, animation, Games are still jealous
  - More like an engineering problem
- Using better, more precise representations than RGB
  - Not used in VFX or Games at all
- Better camera models for everyone!
- Volumetric / Subsurface Scattering effects
- Better BRDF models (maybe “material scanning”)



WOULD YOU LIKE TO KNOW MORE?

Would you like to know more?

- https://solidangle.com
  - http://www.solidangle.com/arnold/research/
- https://unrealengine.com
  - Super cheap, with source code!

Questions?

$$L_o = L_e + \int_{\Omega} L_i \cdot f_r \cdot \cos\theta \cdot d\omega$$